

A note on adjusting R^2 for using with cross-validation

Indrė Žliobaitė^{*1,2} and Nikolaj Tatti²

¹Dept. of Geosciences and Geography, University of Helsinki, Finland

²Helsinki Institute for Information Technology HIIT, Aalto University, Finland

May 6, 2016

Abstract

We show how to adjust the coefficient of determination (R^2) when used for measuring predictive accuracy via leave-one-out cross-validation.

1 Background

The coefficient of determination, denoted as R^2 , is commonly used in evaluating the performance of predictive models, particularly in life sciences. It indicates what proportion of variance in the target variable is explained by model predictions. R^2 can be seen as a normalized version of the mean squared error. Normalization is such that $R^2 = 0$ is equivalent to the performance of a naive baseline always predicting a constant value, equal to the mean of the target variable. $R^2 < 0$ means that the performance is worse than the naive baseline. $R^2 = 1$ is the ideal prediction.

Given a dataset of n points R^2 is computed as

$$R^2 = 1 - \frac{\sum_n (y_i - \hat{y}_i)^2}{\sum_n (y_i - \bar{y})^2}, \quad (1)$$

where \hat{y}_i is the prediction for y_i , and \bar{y} is the average value of y_i . Traditionally R^2 is computed over all data points used for model fitting.

The naive baseline is a prediction strategy which does not use any model, but simply always predicts a constant value, equal to the mean of the target variable, that is, $\hat{y}_i = \bar{y}$. It follows from Eq. (1) that then for the naive predictor $R^2 = 0$.

Cross-validation is a standard procedure commonly used in machine learning for assessing out-of-sample performance of a predictive model [1]. The idea is to partition data into k chunks at random, leave one chunk out from model calibration, use that chunk for testing model performance, and continue the same procedure with all the chunks. Leave-one-out cross-validation (LOOCV) is used when sample size is particularly small, then the test set consists of one data point at a time.

When cross-validation is used, the naive baseline that always predicts a constant value, the average value of the outputs in the training set, gives $R^2 < 0$ if computed according to Eq. 1. This happens due to an improper normalization: the denominator in Eq. 1 uses \bar{y} , and \bar{y} is computed over the *whole* dataset, and not just the training data.

2 Cross-validated R^2

To correct this, we define

$$R_{cv}^2 = 1 - \frac{\sum_n (y_i - \hat{y}_i)^2}{\sum_n (y_i - \bar{y}_i)^2},$$

where \bar{y}_i is the average of outputs without y_i ,

$$\bar{y}_i = \frac{1}{n-1} \sum_{j=1, j \neq i}^n y_j.$$

That is, \bar{y}_i is the naive predictor based on the training data, *solely*.

We show that adjusted R_{cv}^2 for leave-one-out cross-validation can be expressed as

$$R_{cv}^2 = \frac{R^2 - R_{naive}^2}{1 - R_{naive}^2}, \quad (2)$$

^{*}indre.zliobaite@helsinki.fi

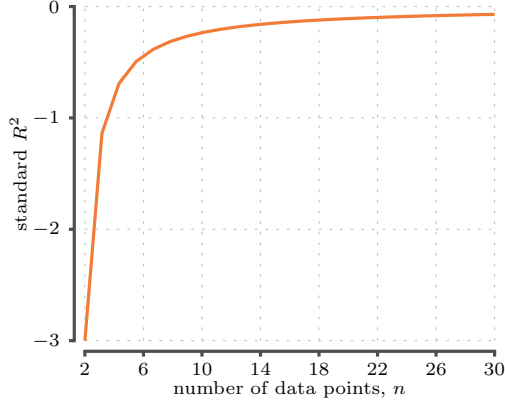


Figure 1: The standard R^2 score for the naive constant predictor.

where R^2 is measured in a standard way as in Eq. (1), and R_{naive}^2 is the result of the naive constant predictor, and is equal to

$$R_{naive}^2 = 1 - \frac{n^2}{(n-1)^2}, \quad (3)$$

where n is the number of data points.

Figure 1 plots the standard R^2 score for the naive predictor, as per Eq. (3).

The remaining part of the paper describes mathematical proof for this adjustment. We will show that R_{naive}^2 does not depend on the variance of the target variable y , only depends on the size of the dataset n .

3 How this works

Let us define R_{naive}^2 as the R^2 score for naive predictor based on training data,

$$R_{naive}^2 = 1 - \frac{\sum (y_i - \bar{y}_i)^2}{\sum (y_i - \bar{y})^2}.$$

Proposition 1. *Let R^2 be the R^2 score of the predictor. The adjusted R^2 is equal to*

$$R_{cv}^2 = \frac{R^2 - R_{naive}^2}{1 - R_{naive}^2}, \quad (4)$$

where the leave-one-out cross-validated R_{naive}^2 for the constant prediction is

$$R_{naive}^2 = 1 - \left(\frac{n}{n-1} \right)^2,$$

where n is the number of data points.

Proof. Let us write

$$A = \sum (y_i - \hat{y}_i)^2, \quad B = \sum (y_i - \bar{y})^2$$

and

$$C = \sum (y_i - \bar{y}_i)^2.$$

Note that $R^2 = 1 - A/B$ and $R_{cv}^2 = 1 - A/C$.

Our first step is to show that $C = \alpha B$, where $\alpha = n^2/(n-1)^2$. Note that A , B and C do not change if we translate $\{y_i\}$ by a constant; we can assume that $n\bar{y} = \sum_{i=1}^n y_i = 0$.

This immediately implies

$$\bar{y}_i = \frac{1}{n-1} \sum_{j=1, j \neq i}^n y_j = \frac{-y_i}{n-1} + n\bar{y} = \frac{-y_i}{n-1}.$$

The i th error term of C is

$$(y_i - \bar{y}_i)^2 = \left(y_i + \frac{y_i}{n-1} \right)^2 = \left(\frac{y_i n}{n-1} \right)^2 = \alpha y_i^2.$$

This leads to

$$C = \alpha \sum_{i=1}^n y_i^2 = \alpha B.$$

Finally,

$$\begin{aligned} \frac{R^2 - R_{naive}^2}{1 - R_{naive}^2} &= \frac{R^2 - 1 + \alpha}{\alpha} \\ &= \frac{1 - A/B - 1 + \alpha}{\alpha} \\ &= 1 - \frac{A}{\alpha B} = 1 - \frac{A}{C} = R_{cv}^2, \end{aligned}$$

which concludes the proof. \square

References

- [1] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.